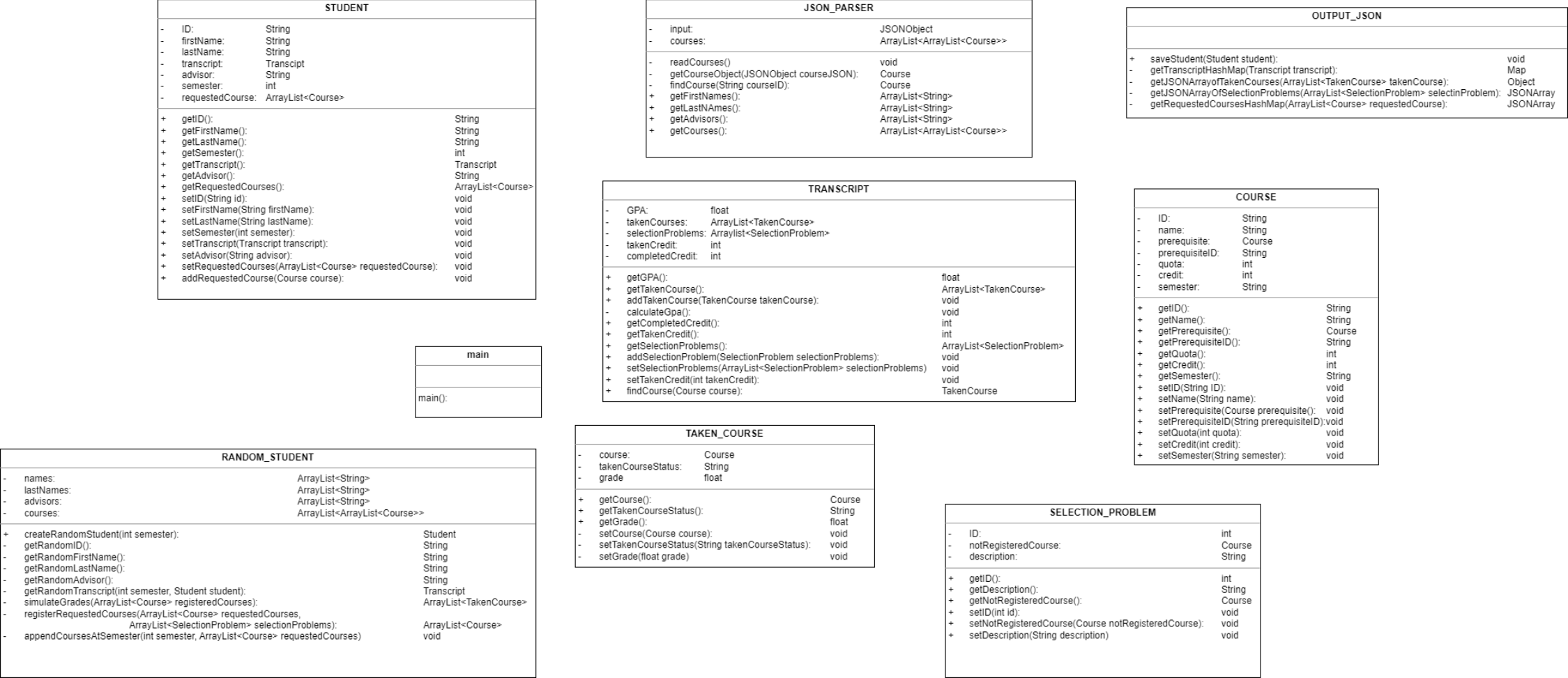
STUDENT
- ID: String
- firstName: String
- lastName: String
- transcript: Transcipt
- advisor: String
- semester: int
- requestedCourse: ArrayList<Course>
+ getID(): String
+ getFirstName(): String
+ getLastName(): String
+ getSemester(): int
+ getTranscript(): Transcript
+ getAdvisor(): String
+ getRequestedCourses(): ArrayList<Course>
+ setID(String id): void
+ setFirstName(String firstName): void
+ setLastName(String lastName): void
+ setSemester(int semester): void
+ setTranscript(Transcript transcript): void
+ setAdvisor(String advisor): void
+ setRequestedCourses(ArrayList<Course> requestedCourse): void
+ addRequestedCourse(Course course): void
JSON_PARSER
- input: JSONObject
- courses: ArrayList<ArrayList<Course>>
- readCourses() void
- getCourseObject(JSONObject courseJSON): Course
- findCourse(String courseID): Course
+ getFirstNames(): ArrayList<String>
+ getLastNAmes(): ArrayList<String>
+ getAdvisors(): ArrayList<String>
+ getCourses(): ArrayList<ArrayList<Course>>
OUTPUT_JSON
+ saveStudent(Student student): void
- getTranscriptHashMap(Transcript transcript): Map
- getJSONArrayofTakenCourses(ArrayList<TakenCourse> takenCourse): Object
- getJSONArrayOfSelectionProblems(ArrayList<SelectionProblem> selectinProblem): JSONArray
- getRequestedCoursesHashMap(ArrayList<Course> requestedCourse): JSONArray
TRANSCRIPT
- GPA: float
- takenCourses: ArrayList<TakenCourse>
- selectionProblems: Arraylist<SelectionProblem>
- takenCredit: int
- completedCredit: int
+ getGPA(): float
+ getTakenCourse(): ArrayList<TakenCourse>
+ addTakenCourse(TakenCourse takenCourse): void
- calculateGpa(): void
+ getCompletedCredit(): int
+ getTakenCredit(): int
+ getSelectionProblems(): ArrayList<SelectionProblem>
+ addSelectionProblem(SelectionProblem selectionProblems): void
+ setSelectionProblems(ArrayList<SelectionProblem> selectionProblems) void
+ setTakenCredit(int takenCredit): void
+ findCourse(Course course): TakenCourse
COURSE
- ID: String
- name: String
- prerequisite: Course
- prerequisiteID: String
- quota: int
- credit: int
- semester: String
+ getID(): String
+ getName(): String
+ getPrerequisite(): Course
+ getPrerequisiteID(): String
+ getQuota(): int
+ getCredit(): int
+ getSemester(): String
+ setID(String ID): void
+ setName(String name): void
+ setPrerequisite(Course prerequisite(): void
+ setPrerequisiteID(String prerequisiteID):void
+ setQuota(int quota): void
+ setCredit(int credit): void
+ setSemester(String semester): void
main
main():
TAKEN_COURSE
- course: Course
- takenCourseStatus: String
- grade float
+ getCourse(): Course
+ getTakenCourseStatus(): String
+ getGrade(): float
- setCourse(Course course): void
- setTakenCourseStatus(String takenCourseStatus): void
- setGrade(float grade) void
SELECTION_PROBLEM
- ID: int
- notRegisteredCourse: Course
- description: String
+ getID(): int
+ getDescription(): String
+ getNotRegisteredCourse(): Course
+ setID(int id): void
+ setNotRegisteredCourse(Course notRegisteredCourse): void
+ setDescription(String description) void
RANDOM_STUDENT
- names: ArrayList<String>
- lastNames: ArrayList<String>
- advisors: ArrayList<String>
- courses: ArrayList<ArrayList<Course>>
+ createRandomStudent(int semester): Student
- getRandomID(): String
- getRandomFirstName(): String
- getRandomLastName(): String
- getRandomAdvisor(): String
- getRandomTranscript(int semester, Student student): Transcript
- simulateGrades(ArrayList<Course> registeredCourses): ArrayList<TakenCourse>
- registerRequestedCourses(ArrayList<Course> requestedCourses, ArrayList<SelectionProblem> selectionProblems): ArrayList<Course>
- appendCoursesAtSemester(int semester, ArrayList<Course> requestedCourses) void